PERFORMANCE EVALUATION PROTOCOL FOR TEXT, FACE, HAND, PERSON AND VEHICLE DETECTION & TRACKING IN VIDEO ANALYSIS AND CONTENT EXTRACTION (VACE-II)

Submitted to
Advanced Research and Development Activity

Technical Monitors: John Garofolo, Rachel Bowers, Dennis Moellman

by

Rangachar Kasturi, Professor
Phone: (813) 974-3561, Fax: (813) 974 5456
Email: r1k@csee.usf.edu
Dmitry Goldgof, Professor
Padmanabhan Soundararajan, Postdoctoral Fellow
Vasant Manohar, Student
Matthew Boonstra, Student
Valentina Korzhova, Student

Computer Science & Engineering
University of South Florida, ENB 118
4202 E. Fowler Ave
Tampa, FL 33620-5399

Date: Oct 12, 2005

Contents

List of Figures

List of Tables

1 Introduction

This evaluation focuses on a set of technologies to detect and track specific object types in video. These technologies are termed *core* technologies since it is believed that they will form the basis for a variety of useful extraction applications and will serve as important low-level components for higher-level event recognition technologies. While these technologies are not necessarily *atomic* from a technology point of view, the tasks they seek to address are relatively atomic in-terms of human annotation. Thus, a balance has been struck between issues in creation of reliable reference annotations and core task definitions. This evaluation does not seek to address all core tasks which might be of importance, rather priority is placed on tasks which can be viewed as important to more that one of the challenge domains and which would be critical for several possible application areas.

2 Text Detection Task

The goal of the text detection task is to identify the blocks of text in each frame. These text blocks are to be identified by bounding boxes as defined in the annotation guidelines document [4]. Note that these boxes are to be oriented to the angle of the text as it appears relative to the frame. Text in a frame is to be treated as a single block unless it is separated as specified in the annotation guidelines.

Since this is a frame-based task, the performance of the task will be scored at the frame level and will be based on how accurate the system output boxes align with respect to the ground truth. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Sec 12.3. Text which is annotated as unevaluable by the annotators will not be evaluated. The performance of the task will be scored using the Sequence Frame Detection Accuracy measure described in Sec 15.1. The scoring will be broken down by the readability levels annotated in the reference.

For this particular task, these tags will include:

1. Video filename.
2. Object id (unique for the frame).
3. BBox location parameters upper left corner, height, width and rotation attribute if needed.
4. If BBox size and position remains constant, can include this as a framespan.
5. Special exception tags in the reference:
   (a) DCF if *crowded* text.
   (b) DCR if text present and unreadable.

3 Text Tracking Task

The goal of the text tracking task is to identify the blocks of text in each frame and track them throughout the given sequence. This is similar to the text detection task. However, the ID assigned by the system to each detected object must be used to uniquely identify the object across all frames in the sequence. As with the text detection task, the text blocks are to be identified by bounding boxes as defined in the annotation guidelines document [4] and the same rules apply to the generation of the boxes as the text detection task. However, once a system identifies an object, it must be correctly identified with the same Object ID across frames to be scored as correct.

Since this is a sequence-based task, the scoring will be performed at the sequence level. The performance of the task will be scored using the Sequence Tracking Detection Accuracy Measure described in Sec 15.2. The scoring will be broken down by the readability levels annotated in the reference.

For this particular task, these tags will include:

1. Video filename.
2. Object id (unique for the sequence).
3. BBox location parameters for the upper left corner, height, width and rotation attribute if needed.
4. If BBox size and position remains constant, can include this as a framespan.
5. Special exception tags in the reference:
   (a) DCF if *crowded* text.
   (b) DCR if text present and unreadable.

4 Face Detection Task

The goal of the face detection task is to identify the faces in each frame. The faces are to be identified by bounding boxes as defined in the annotation guidelines document [4]. Note that these boxes are to be oriented to the angle of the face as it appears relative to the frame.

Since this is a frame-based task, the performance of the task will be scored at the frame level and will be based on how accurate the system output boxes align with respect to the ground truth. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Sec 12.3. Faces which are annotated as unevaluable by the annotators will not be evaluated. The performance of the task will be scored using the Sequence Frame Detection Accuracy measure described in Sec 15.1.

For this particular task, these tags will include:

1. Video filename.
2. Object id (unique for the frame).
3. BBox location parameters for the upper left corner, height and width
4. If BBox size and position remains constant, can include this as a framespan.
5. Special exception tags in the reference:
   (a) DCF if *crowded* faces.
   (b) DCR if face is present but not seen clearly.
   (c) DCO if *synthetic face*.

5 Face Tracking Task

The goal of the face tracking task is to identify the faces in each frame and track them throughout the given sequence. This is similar to the face detection task. However, the ID assigned by the system to each detected object must be used to uniquely identify the object across all frames in the sequence. As with the face detection task, the face are to be identified by bounding boxes as defined in the annotation guidelines document [4] and the same rules apply to the generation of the boxes as the face detection task. However, once a system identifies an object, it must be correctly identified with the same Object ID across frames to be scored as correct.

Since this is a sequence-based task, the scoring will be performed at the sequence level. The performance of the task will be scored using the Sequence Tracking Detection Accuracy Measure described in Sec 15.2.

For this particular task, these tags will include:

1. Video filename.

2. Object id.
3. BBox location parameters for the upper left corner, height and width
4. If BBox size and position remains constant, can include this as a framespan.
5. Special exception tags in the reference:
   (a) DCF if *crowded* faces.
   (b) DCR if face is present but not seen clearly.
   (c) DCO if *synthetic face*.

6 Hand Detection Task

The goal of the hand detection task is to just identify the hands as a part of a person. The hand will be identified by a *point* as defined in the annotation guidelines document [4]. Since this is a frame-based task, the performance of the task will be scored at the frame level and the distance based measure (SFDA-D) as described in Section 14.1 will be used. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Section 12.3. For this particular task and representation, these tags will include:

1. Video filename.
2. Object ID (unique for the frame).
3. Location of the point.
4. Special exception tags in the reference:
   (a) DCF if crowd.
   (b) DCR if hands present but not clear.

7 Hand Tracking Task

The goal of the hand tracking task is to identify and track the hand in a sequence. Since this is a sequence based task, the performance of the task will be scored at the sequence level and be based on how close the position of the system output is with respect to the ground truth consistently in all frames in the sequence. Since this is a sequence based task, the performance will be scored at the sequence level and the distance based measure (STDA-D) as described in Section 14.2 will be used. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Section 12.3. For this particular task and representation, these tags will include:

1. Video filename.
2. Object ID (unique for the frame).
3. Location of the point.
4. Special exception tags in the reference:
   (a) DCF if crowd.
   (b) DCR if hand present but not clear.

8 Person Detection Task

The goal of the person detection task is to identify the person in each frame. The person will be identified and evaluated based on the resolution of the data at hand. In other words, the definition of a person is not the same in different domains, the specific conditions of which can be referred to in the annotation guidelines document [4].

8.1 Box Definition

In domains where the resolution of the person is detailed enough, the person will be identified keeping into account the head and torso aspects of each person while annotating. Refer to the annotation guidelines document [4] for specific annotation details about how the head area and the torso area is marked. This annotation document will also specify when this definition applies to annotate the person. The reason to choose the Box definition is that for this year we decided arbitrarily that the ellipse shape can model the shape of head and the Box shape can model the head, torso and leg regions as a whole. Additionally the annotation complexity is also reduced as opposed to annotating individual body parts of the person. For the evaluation since this is at a frame level and is an approximation of area, the SFDA area based measure as described in Section 15.1 will be used. The scoring is proportional to the area overlap between the system output and the ground truth. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Section 12.3. For this particular task and representation, these tags will include:

1. Video filename.
2. Object ID (unique for the frame).
3. Box parameters.
4. Special exception tags in the reference:
   (a) DCF if crowd.
   (b) DCR if person present but very unclear.

8.2 Point Definition

In domains (for example, UAV) where the resolution is low, the person will be identified by a point (or a minimal bounding box) and will be evaluated using the SFDA-D distance based measure as described in Section 14.1. The scoring is inversely proportional to the distance between the system output and the ground truth. For exact details on when this definition will be used to annotate the person refer to the annotation guidelines [4] document. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Section 12.3. For this particular task and representation, these tags will include:

1. Video filename.
2. Object ID (unique for the frame).
3. Location of the point.
4. Special exception tags in the reference:
   (a) DCF if crowd.
   (b) DCR if person present but very unclear.

9 Person Tracking Task

The goal of the person tracking task is to identify and track the person in a sequence. The person will be identified and evaluated based on the resolution of the data at hand. In other words, the definition of a person is not the same in different domains, the specific conditions of which can be referred to in the annotation guidelines document [4].

9.1 Box Definition

In domains where the resolution of the person is detailed enough, the person will be identified keeping into account the head and torso aspects of each person while annotating. Refer to the annotation guidelines document [4] for specific annotation details about how the head area and the torso area is marked. This annotation document will also specify when this definition applies to annotate the person. The reason to choose the Box definition is that for this year we decided arbitrarily that the ellipse shape can model the shape of head and the Box shape can model the head, torso and leg regions as a whole. Additionally the annotation complexity is also reduced as opposed to annotating individual body parts of the person. For the evaluation since this is at the sequence level and is an approximation of area, the STDA area based measure as described in Section 15.2 will be used. The scoring is directly proportional to the area overlap between the system output and the ground truth. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Section 12.3. For this particular task and representation, these tags will include:

1. Video filename.
2. Object ID (unique for the frame).
3. Box parameters.
4. Special exception tags in the reference:
   (a) DCF if crowd.
   (b) DCR if person present but very unclear.

9.2 Point Definition

In domains (for example, UAV) where the resolution is low, the person will be identified by a point (or a minimal bounding box) and will be evaluated using the STDA-D distance based measure as described in Section 14.2 will be used. The scoring is inversely proportional to the distance between the system output and the ground truth. For exact details on when this definition will be used to annotate the person refer to the annotation guidelines [4] document. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Section 12.3. For this particular task and representation, these tags will include:

1. Video filename.
2. Object ID (unique for the frame).
3. Location of the point.
4. Special exception tags in the reference:
   (a) DCF if crowd.
   (b) DCR if person present but very unclear.

10 Vehicle Detection Task

The goal of the vehicle detection is to identify the vehicle in each frame. The vehicle will be identified and evaluated based on the resolution of the data at hand. In other words, the definition of a vehicle is not the same in different domains, the specific conditions of which can be referred to in the annotation guidelines document [4].

10.1 Box Definition

If the annotation specification defines a Bounding box for the vehicles, then the performance will be evaluated by the area based SFDA measure described in Section 15.1. The exact details on when this definition of representation will be used for annotation refer to the annotation guidelines document [4]. The scoring is directly proportional to the area overlap between the system output and the ground truth. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Section 12.3. For this particular task and representation, these tags will include:

1. Video filename.
2. Object ID (unique for the frame).
3. BBox parameters.
4. Special exception tags in the reference:
   (a) DCF if crowd of vehicles.
   (b) DCR if vehicle is present but not clear.

10.2 Point Definition

If the annotation specification defines a point definition for the vehicles then the performance will be evaluated by the distance based SFDA-D measure described in Section 14.1. The exact details on when this definition of representation will be used for annotation, refer to the annotation guidelines document [4]. The scoring is inversely proportional to the distance between the system output and the ground truth. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Section 12.3. For this particular task and representation, these tags will include:

1. Video filename.
2. Object ID (unique for the frame).
3. Location of the point.
4. Special exception tags in the reference:
   (a) DCF if crowd of vehicles.
   (b) DCR if vehicle present but not clear.

11 Vehicle Tracking Task

The goal of the vehicle tracking task is to identify and track the vehicle in a sequence. The vehicle will be identified and evaluated based on the resolution of the data at hand. In other words, the definition of a vehicle is not the same in different domains, the specific conditions of which can be referred to the annotation guidelines document [4].

11.1 Box Definition

If the annotation specification defines a Bounding box for the vehicles, then the performance will be evaluated by the area based STDA measure described in Section 15.2. The exact details on when this definition of representation will be used for annotation refer to the annotation guidelines document [4]. The scoring is directly proportional to the area overlap between the system output and the ground truth. The system output tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Section 12.3. For this particular task and representation, the tags will include:

1. Video filename.
2. Object ID (unique for the frame and sequence).
3. BBox parameters.
4. Special exception tags in the reference:
   (a) DCF if crowd of vehicles.
   (b) DCR if vehicle is present but not clear.

11.2 Point Definition

If the annotation specification defines a point definition for the vehicles then the performance will be evaluated by the distance based STDA-D measure described in Section 14.2. The exact details on when this definition of representation will be used for annotation, refer to the annotation guidelines document [4]. The scoring is inversely proportional to the distance between the system output and the ground truth in the entire sequence. The system out tags must be generated according to the rules specified in the annotation guidelines and are to be formatted as described in Section 12.3. For this particular task and representation, these tags will include:

1. Video filename.
2. Object ID (unique for the frame).
3. Location of the point.
4. Special exception tags in the reference:
   (a) DCF if crowd of vehicles.
   (b) DCR if vehicle present but not clear.

12 Scope

The dataset includes data from the meeting room, broadcast news, Surveillance and UAV domains.

12.1 VACE-II 2004-2005 Datasets

This section describes the dataset to be developed to support the 2004-2005 evaluations. A complete set of training and test data that will be supported for each task and domain are shown in Table 1 for the data statistics which include the planned breakdown for the Micro Corpus, Training and Evaluation data. The number of sequences and times shown are estimates and could change based on data availability and annotation complexity.

For the 2005 evaluations, given the available resources and time, the following core tasks/domain will be supported with annotated data as indicated in Table 2.

	DATA	NUMBER OF SEQUENCES	TOTAL MINUTES	AVERAGE MINUTES PER SEQUENCE
PER DOMAIN	MICRO-CORPUS	5	10	–
	TRAINING	50	175	2.5
	EVALUATION	50	175	2.5

Table 1: VACE-II Corpus Partitioning for each Core Eval Task.

TASK	**DOMAIN**			
	Meeting Room NIST Meeting Room Project	Broadcast News LDC Broadcast (ABC, CNN & Al-Jazeera)	UAV Vivid-II	Surveillance *Pending availability*
Hand Detect & Track	Y	Y	–	–
Person Detect & Track	Y	Y	Y	Y
Vehicle Detect & Track	–	Y	Y	Y

Table 2: Task versus Domain Support Matrix (– = No, Y = Yes, ? = Unsure).

12.2 Permitted Side Information

The following information for each domain will be available to the systems in performing the tasks. No other side information should be used.

- Broadcast News domain:
- Camera information (if available)
    1. Channel information
    2. Year
    3. Language used
- Meeting Room domain:
    1. **Required primary -** no side information is available but developers can make any assumptions from the training data (What kind of camera, Pan, zoom, etc)
    2. **Adaptation Contrast -** over the entire evaluation sequence.
    3. **Manual initialization -** permitted on first 10 seconds of each test clip **(Manual contrast)**: this would mean that the first 10 seconds will not be used in the evaluation score generations.
- UAV domain:
    1. **Contrast -** Fusion of multiple streams.

The training procedures used in all three conditions must be clearly document in your system description for each run. If the systems use any of the contrastive side information, they **must** report it appropriately (i.e, as a contrastive condition). Refer Sec 19 for more details

12.3 Formats

As an expedient for this year the ViPER native format will be used for both the system output and reference annotations. Both the input and output files will contain the tags required for evaluation. An example XML file produced by ViPER can be found in the annotation guidelines document [4].

13 Performance Assessment

This section and the following sections will address how the output of the research systems will be evaluated.

For the VACE-II person and vehicle evaluations, we have defined performance measure specific for each domain. In domains where a point definition is defined, the distance based comprehensive measures described in Sections 14.1– 14.2 are used. In domains where the Bounding Box definitions are used in annotations, the area based comprehensive measures described in Sections 15.1– 15.2 are used. These will be the primary measures for evaluations and they will provide not only a summative measure of the performance of the systems, but they will also provide the researchers with a focused tool to use in developing and improving their systems.

Before proceeding further, let us define the terms we will use in describing the performance measures:

1. **Object** - the entities of interest (e.g. text, vehicles, faces, etc.)
2. **Object class** - a constrained set of objects (e.g. caption text, school bus, etc.)
3. **Output box** - a geometric shape produced by an algorithm as a result of detection
4. **Measure** - a formula for measuring an algorithm's performance after an experiment

13.1 Handling Limitations in the Ground Truth

Sometimes we want to exclude certain frames from evaluation because they contain frame-level events which place them outside of the scope of the task. An example of this is that the existence of a crowd of faces in a sequence of frames precludes the annotation of particular faces during those frames for the face detection task. To address this issue, **Don't Care Frames (DCFs)** will be established prior to scoring the test results using information in the reference annotation. In our face detection example, particular frames would be annotated in the reference as containing crowds and would not contain further facial annotations. These frames would need to be excluded from evaluation for the face detection task. The DCFs for each task will be automatically generated using a set of rules applied to the reference annotations for that task. Frames in both the reference and system output which are designated as DCFs will then be automatically ignored by the scoring procedure.

Likewise, sometimes we want to exclude certain objects from the target object class because they contain attributes which place them outside the scope of the task. An example of this is the existence of a synthetic face (cartoon or painting) in a particular frame for the face detection task. To address this issue, **Don't Care Objects (DCOs)** will be established prior to scoring the test results using information in the reference annotations. In our synthetic face example, a face annotated as being synthetic would participate in the one-to-one reference/system-output alignment procedure for the new comprehensive measures, but would not be scored. Therefore, an algorithm would not be penalized for missing the synthetic face, but would also not be rewarded for detecting it. Objects in these DCOs will be effectively treated as not existing in both the reference and system output. Additional secondary diagnostic scoring runs may be made to indicate how well these out-of-scope objects were detected/tracked by turning off certain DCOs[1].

Where **DCOs** are used to annotate objects which can be spatially annotated but which can't be reliably identified, some objects may be too blurry or too difficult to localize and cannot be bounded. To address this problem, **Don't Care Regions (DCRs)** will be used to identify areas in frames which can't be spatially annotated and which are to be eliminated entirely from the mapping and scoring process. Detected objects which fall inside a **DCR** or whose area is contained primarily within a **DCR** will be eliminated prior to the mapping/scoring process and will thus not generate false alarm errors. An example is a region of completely unreadable text which can't be effectively grouped into text boxes.

For all the VACE measures, we assume that the **DCF**s and **DCO**s have been removed from both the ground-truth and the algorithm's output prior to the scoring process.

[1] An additional example of a DCO is text classified as of poor quality for the text detection and tracking tasks.

13.2 Handling Spatial Inconsistencies in the Ground Truth

While the annotation guidelines will be made as specific as possible for each task, it is understood that there will be some variability in the generation of the ground truth bounding boxes/distance by the human annotators. This variability will be measured by examining the portion of the data that is doubly annotated for each task. The resulting variance will be used to determine a spatial variance for each ground truth bounding box/distance. Computing the variance is a two step process. First one of the annotator's output is scored against the *other* annotator's output as the ground truth. This scoring is repeated by swapping the annotator's output. The variance is the resulting difference between the two scores. This variance would then determine the level of confidence of the final score.

14 Performance Measures for Point Definitions of Hand, People and Vehicle

This section describes the performance measures where the object definitions are defined by a point. These set of measures are distance based.

14.1 Sequence Frame Detection Accuracy–Distance (SFDA-D) for Frame based Detection Task

The **Sequence Frame Detection Accuracy–Distance** (SFDA-D) makes the distinction between the individual objects in the frame and requires a unique one-to-one mapping of the ground truth and detected objects using some optimization [2]. The mapping will be performed so as to maximize the measure score.

14.1.1 Formula

This is a location based measures which rewards accuracy while penalizing fragmentation. We refer to the ground truth object centroid as (cg_x^i, cg_y^i), where cg_x^i refers to the x co-ordinate of the i^{th} ground truth object. Similarly the system output centroid can be referred to as (co_x^i, co_y^i). We can compute Euclidean distances between the ground truth and the system output objects and call this the **D** matrix. Note that this matrix need not be a square matrix as there can be unequal number of system outputs and ground truth objects.

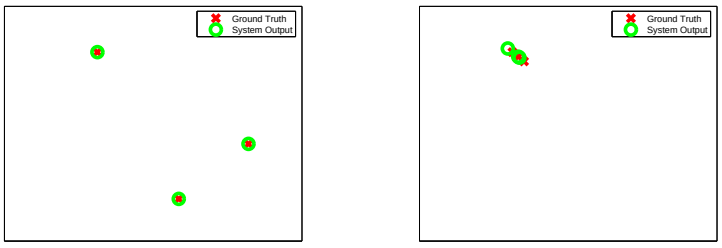
Additionally this **D** matrix is normalized by a quarter of the maximal possible distance between any two objects in the frame *viz., the diagonal*. We can use the **D** matrix as the input to the assignment problem where the algorithm will give us the mapped object sets.

Using the assignment sets, we can compute for each frame t, the Frame Detection Accuracy–Distance (FDA-D) as,

$$FDA-D(t) = \frac{\sum_{i=1}^{N_{mapped}^{(t)}} (1 - d_i')}{\left\lceil \frac{N_G^{(t)} + N_D^{(t)}}{2} \right\rceil} \quad (1)$$

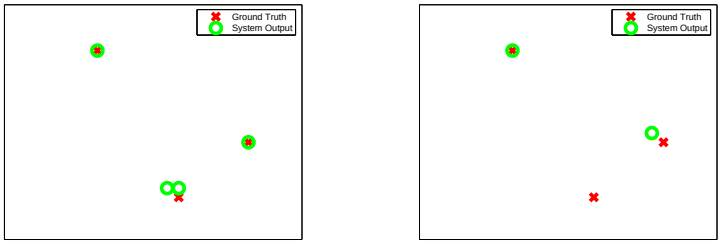
where, N_{mapped} is the number of mapped object sets and d_i' represents the distance between the i^{th} mapped pair. Examples are shown in Figs 1– 2.

[2] Potential strategies to solve this assignment problem are the weighted bi-partite graph matching and the Hungarian algorithm. The one-to-one mapping has many issues ranging from whether it is feasible to perform this in a reasonable amount of time to its capability in capturing the accuracy. Another complicated alternative is to consider one-to-many mapping which will require more computational time but has potential advantages. With one-to-many mapping, each ground truth object can be matched to multiple detected object (this requires complicated book-keeping) which after proper post-processing will result in optimal matching. Also, one-to-many will penalize fragmentation both in the spatial as well as spatio-temporal dimension depending on the specific measure used in the strict minimum order sense. In this evaluation to keep things manageable, we propose to use the one-to-one mapping. More details of the strategies can be found in Appendix A.



(a) (b)

Figure 1: (a) FDA–D = 1.0, System Output is perfectly aligned with the Ground truth objects. (b) FDA–D = 0.958, Special case where the ground truth is close-by and system outputs are not aligned properly.



(a) (b)

Figure 2: (a) FDA–D = 0.825, False Alarms (b) FDA–D = 0.737, Missed Detects.

Further, this can then be averaged over the set of all frames in a sequence by,

$$SFDA-D(s) = \frac{\sum_{t=1}^{N_{frames}} FDA-D(t)}{\sum_{t=1}^{N_{frames}} \exists(N_G^{(t)} \; OR \; N_D^{(t)})} \tag{2}$$

and for the average over all the set of sequences we have,

$$ASFDA-D = \frac{\sum_{s=1}^{S} SFDA-D(s)}{S} \tag{3}$$

where, S = total number of sequences.

14.2 Sequence Tracking Detection Accuracy–Distance (STDA-D) Measure for Object based Tracking Task

The **Sequence-based Tracking Accuracy Measure** (STDA-D) is based on how well a system can detect and track the person in the entire sequence. This measure will penalize fragmentation in the spatio-temporal space.

14.2.1 Formula

$$STDA-D = \sum_{i=1}^{N_{mapped}} \frac{\sum_{t=1}^{N_{frames}} (1-d_t')}{N_{(G_i \cup D_i \neq \emptyset)}} \tag{4}$$

using similar notations as in Section 14.1.
We define,

$$ATA-D = \frac{STDA-D}{\frac{N_G+N_D}{2}} \tag{5}$$

The ATA–D is the average of the STDA–D measure over all the objects in the sequence. TO measure the performance over all the sequences, we will use the Average ATA-D (AATA–D) measure, which is the average of the ATA–D measure over all the sequences in the test set.

$$AATA-D = \frac{\sum_{i=1}^{S} ATA-D(i)}{S} \tag{6}$$

where, S is the total number of sequences considered.

15 Performance Measures for Box Definitions of People and Vehicle

This section describes the performance measures where the object definitions are for a person or vehicle with bounding box definitions and these measures are area-based.

15.1 Sequence Frame Detection Accuracy (SFDA) Measure for Frame Based Detection Task

The purpose of the **Sequence Frame Detection Accuracy** measure is to assess the algorithm's detection accuracy. The measure provides an objective function for the accuracy of the system with regard to several factors including temporal, spatial and number of objects. The comprehensive measure described below will be the primary measure for the evaluation. This is an accuracy metric and produces a real number value between zero (worst possible performance) and one (best possible performance).

The **Sequence Frame Detection Accuracy** measure makes the distinction between the individual objects in the frame and requires a unique one-to-one mapping of ground truth and detected objects using some optimization The mapping will be performed so as to maximize the measure score.

15.1.1 Formula

This is an area-based measure which penalizes false detections, missed detections and spatial fragmentation. For a single frame t, we define **FDA(t)** as the frame detection accuracy, given that there are N_G ground-truth objects and N_D detected objects in the t^{th} frame as,

$$FDA(t) = \frac{\text{Overlap Ratio}}{\frac{N_G^{(t)}+N_D^{(t)}}{2}} \tag{7}$$

$$\text{where, Overlap Ratio} = \sum_{i=1}^{N_{mapped}} \frac{|G_i^{(t)} \bigcap D_i^{(t)}|}{|G_i^{(t)} \bigcup D_i^{(t)}|} \tag{8}$$

Here, the N_{mapped} is the number of mapped objects.

The **FDA** measure will be evaluated over the entire sequence and we can refer to this as the sequence frame detection accuracy (**SFDA**), which can be defined as the ratio of the sum total of the **FDA** over the sequence to the number of frames in the sequence where either the ground-truth or detected box exists. In simpler terms, this is the average of the **FDA** measure over all the frames in the sequence. This can be expressed as,

$$SFDA = \frac{\sum_{t=1}^{t=N_{frames}} FDA(t)}{\sum_{t=1}^{t=N_{frames}} \exists(N_G^{(t)} \ OR \ N_D^{(t)})} \tag{9}$$

This **SFDA** measures the **FDA** over the considered sequence exclusively. To measure the performance over all the sequences, we will use the Average Sequence Frame Detection Accuracy (**ASFDA**). In simpler terms this is the average **SFDA** over-all the sequences in the test set and can be expressed as,

$$ASFDA = \frac{\sum_{i=1}^{S} SFDA(i)}{S} \tag{10}$$

where, S is the total number of sequences considered.

15.1.2 Relaxing Spatial Inaccuracies

While the system outputs might not align well with the annotator box, there are various methods in which we can give credit for spatial inconsistencies. To make the explanations easier, we present here a series of comparative examples where the SFDA scores are computed subjected to different thresholding options. Fig 3 shows an example on a particular frame. There are 3 ground truth boxes and 2 annotator boxes. The FDA scoring and the individual scoring is indicated below in the same figure.

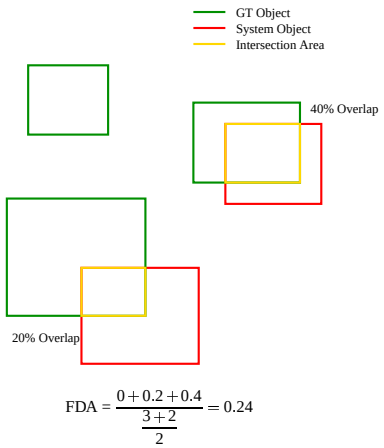


Figure 3: FDA Score without any thresholding

1. **Non-binary Decision Thresholding:**

   The thresholding is as defined in Eq 11 and Eq 12. Fig 4 shows an example with this style of thresholding.

$$\text{Thresholded Overlap Ratio} = \frac{FDA\_T\_NB}{|G_i^{(t)} \cup D_i^{(t)}|} \tag{11}$$

   where,

$$FDA\_T\_NB = \begin{cases} |G_i^{(t)} \cup D_i^{(t)}|, & \text{if } \frac{|G_i^{(t)} \cap D_i^{(t)}|}{|G_i^{(t)} \cup D_i^{(t)}|} \geq THRESHOLD \\ |G_i^{(t)} \cap D_i^{(t)}|, & otherwise \end{cases} \tag{12}$$

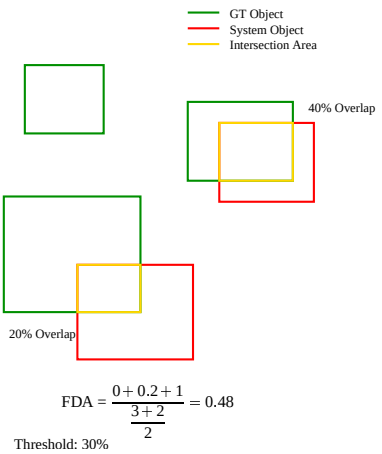


Figure 4: FDA Score with non-binary thresholding (thresholded at 30%)

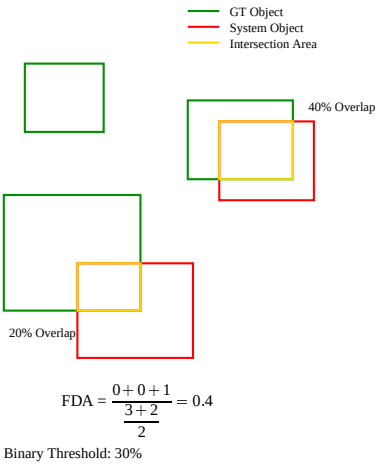


Figure 5: FDA Score with binary thresholding (thresholded at 30%)

2. **Binary Decision Thresholding:**

   The thresholding is as defined in Eq 13 and Eq 14. Fig 5 shows an example with this style of thresholding.

$$\text{Thresholded Overlap Ratio} = \frac{FDA\_T\_B}{|G_i^{(t)} \cup D_i^{(t)}|} \tag{13}$$

where,

$$FDA\_T\_B = \begin{cases} |G_i^{(t)} \cup D_i^{(t)}|, & \text{if } \frac{|G_i^{(t)} \cap D_i^{(t)}|}{|G_i^{(t)} \cup D_i^{(t)}|} \geq THRESHOLD \\ 0, & otherwise \end{cases} \tag{14}$$

15.2 Sequence Tracking Detection Accuracy (STDA) Measure for Object Based Tracking Task

In tracking tasks, systems must detect objects of interest and follow them through a sequence of frames. As with detection, systems must identify spatial locations of target objects in each frame. However, in the case of tracking, the definition of an object extends over time rather than just a single frame. While many forms of tracking can be imagined, for the purposes of this evaluation, tracking will consist of simply identifying detected objects across contiguous frames. The task is similar to detection, with detected objects linked by a common identity across frames. For the purpose of this evaluation, objects which leave the frame and return later in the sequence need not be identified as the same object. However, occluded objects are to be treated as the same object. But tracking is optional during the occlusion.

Like detection this is a spatio-temporal task and its performance can be assessed with a measure similar to the Sequence Frame Detection Accuracy measure described in Section 15.1. The significant difference between the measures is that in detection tasks the mapping between the system output and reference annotation objects is optimized on a frame-by-frame basis, whereas for tracking, the mapping is optimized on a sequence basis and objects with the same identity cannot be mapped to different objects in different frames in the same track. One of the advantages of making these tasks highly parallel to the detection tasks is that the Sequence Frame Detection Accuracy measure can also be applied to the tracking output to quantify the degradation in tracking due to mis-identification of objects across frames.

15.2.1 Formula

This is a spatio-temporal based measure which penalizes fragmentation in both the temporal as well as the spatial aspect of the measurement. This is an accuracy metric and produces a real number value between zero (worst possible performance) and one (best possible performance). The algorithm is required to output unique IDs for each object in the segment.

In VACE-II, systems will not be evaluated on their object recognition capabilities. This implies that, systems need not identify an object (i.e. assign the same ID to an object) once it gets occluded and reappears at a later instance in the sequence. An object is said to be occluded in a frame if the minimal set of features required to declare its presence is not apparent in that frame. For instance, for a face to be declared in a frame, at least one eye, nose and part of the mouth should be seen.

Let us define objects here to be present in a sequence of frames. G_i denotes the i^{th} ground truth object and $G_i^{(t)}$ denotes the i^{th} ground truth object in t^{th} frame. D_i denotes the detected object for G_i. N_G denotes the number of ground truth objects to be tracked. Let N_{frames} be the total number of ground truth frames in the sequence. Say that for a sequence we have N_G ground truth-ed objects and N_D detected objects. We assume that there exists a 1-1 matching between these objects (the matching strategy itself is performed by specifically computing the measure over all the ground truth and detected object combinations and to maximize the overall score for a sequence) and the **STDA** can be computed by the expression shown below,

$$STDA = \sum_{i=1}^{N_{mapped}} \frac{\sum_{t=1}^{N_{frames}} \left[\frac{|G_i^{(t)} \cap D_i^{(t)}|}{|G_i^{(t)} \cup D_i^{(t)}|} \right]}{N_{(G_i \cup D_i \neq \emptyset)}} \tag{15}$$

Analyzing the numerator part of this equation, *viz*

$$\text{TDA} = \sum_{t=1}^{N_{frames}} \frac{|G_i^{(t)} \cap D_i^{(t)}|}{|G_i^{(t)} \cup D_i^{(t)}|} \tag{16}$$

This is merely the overlap of the detected boxes over the ground truth frame t in the sequence. Observe that this expression is very similar to the Overlap Ratio in Eq 8. Dividing the term by $|G_i^{(t)} \cup D_i^{(t)}|$ and summing over all detected boxes and ground truth boxes, we make sure that we penalize for both False Negatives (undetected ground truth area) and False Positives (detected boxes that do not overlay any ground truth area) and this gives us a raw estimate of the measure.

Observe the summation term in Eq 16, the summation over all the frames. This is the the part of the expression which is influenced by the ability of an algorithm to track.

The normalization of the **TDA** expression (the denominator) denoted by the $N_{(G_i \bigcup D_i \neq \emptyset)}$, which indicates the total number of frames in which either a ground truth object or a detected object or both are present. This **STDA** is the measure over all the objects in the sequence. The summation runs from 1 through to N_{mapped} which indicates the number of mapped objects.

We define,

$$ATA = \frac{STDA}{\frac{N_G + N_D}{2}} \tag{17}$$

In simpler terms, this is the average of the **STDA** measure over all the objects in the sequence.

The numerator in the expression rewards true positives and penalizes false alarms. This provides a numeric value of the percentage of ground truth that is covered with mapped objects[3].

To measure the performance over all the sequences, we will use the Average ATA, **AATA** measure which is the average of the ATA measure over all the sequences in the test set.

$$AATA = \frac{\sum_{i=1}^{S} ATA(i)}{S} \tag{18}$$

where, S is the total number of sequences considered.

The issue of fragmentation is also critical. This measure will penalize the detected boxes which fragments an object block. In a scenario where the algorithm detects say two objects while the ground truth has only one object, the new measure will penalize this algorithm as opposed to an algorithm which does not fragment. The fragmentation is not addressed in the **STDA** measure but is in the **ATA** measure. Specifically, the fragmentation penalty factor from the **ATA**, will be helpful when we consider the overlap thresholding approach (See Section 15.2.2).

15.2.2 Relaxing Spatial Inaccuracies in Tracking

In order to examine the ability of the algorithms to track objects across frames without regard to the spatial location, it maybe desirable to measure the cross frame tracking aspect and not be concerned with the spatial component within a frame. In this case, we can relax the spatial component penalty by using an area thresholded approach. Please refer to Sec 15.1.2

If in implementing the evaluations we find that exact spatial matching is extremely difficult or impossible, we may adjust this parameter to normalize the test results to more meaningful values.

The threshold ideally would be the variance of the annotator ground truths, which will be computed as described in Section 13.2. We could also compute the final scores for different threshold values. Also, if the annotator variance is taken into consideration at the spatial level here, it will not be accounted for in the final scoring again.

16 Reference Annotations

The Video Performance Evaluation Resource (ViPER) [1] was developed as a tool for ground-truthing video sequences and will be used to create the reference annotations for this evaluation. Objects are marked by bounding box parameters. The objects are annotated in ViPER XML format. The ground truth annotation instructions for all the tasks can be found in the companion annotation guidelines document [4].

[3] If a sequence length dimension is needed, then we could perform the measurements on varying sequence lengths. For this we propose a MIN_SEQUENCE_LENGTH and a MAX_SEQUENCE_LENGTH and could perform analysis of different vendor algorithms in specific ranges. The idea here is to categorize the strengths and weaknesses of the algorithms, relatively. A MIN_SEQUENCE_LENGTH of 1 minute is proposed and MAX_SEQUENCE_LENGTH is a variable parameter but 4 minutes is a viable option.

17 System Input/Output

The system output is to be in ViPER XML format using the tags specified in the task definitions. Note that, the reference will be richly annotated with a variety of information some of which is intended for data selection and analysis only. Therefore, not all the annotated information will be used for evaluation. The proposed file naming conventions are as shown below,

FILENAME EXTENSION	DESCRIPTION
*.gtf	Ground Truth File
*.rdf	Result File
*.ndx	Index File
*.sysinfo	System Information File

Table 3: File naming conventions.

17.1 System Input Data (Training/Testing)

The input data will be in MPEG-2 format as indicated earlier. The data will be presented to the research systems in multiple sequences varying in duration from 1–4 minutes. The video clips for each task will be present in a separate directory. An index file will exist for each task and will follow the naming conventions as explained below.

Year_Purpose_Domain_Task.ndx

where,

- *Year* specifies the year in which the evaluation would take place
- *Purpose* can be (Train, Test)
- *Domain* can be (Meetings, BNews, Surveillance, UAV)
- *Task* can be (HD, HT, PD, PT, VD, VT)

Also, the index file will contain the following details.

- Sequence-ID
- Source Path/Filename
- Begin-frame
- End-frame
  where,
  - Sequence-ID is the input sequence ID which can take values (1 ... N_{seq})
  - Source Path/Filename is the path and filename of the original file from which the clip was extracted
  - Begin-frame is the frame number in the original source file when the clip begins
  - End-frame is the frame number in the original source file when the clip ends

Thus, together with the index filename and the information present in the file, we can uniquely identify a video clip and its original source file. Based on the *Sequence-ID*, we can map back to the original file with the details in the corresponding index file. The ground truth XML file will be present in the same directory, with the following naming convention.

Year_Purpose_Domain_Task_Sequence-ID.gtf

Each individual XML file will contain a header listing the tags used in the file and their possible values. For convenience a copy of the XML headers will be included in a separate *config* file.

17.2 System Output Data

The primary submission from each site should use the equal error rate operating point setting for each algorithm/task combination.

The system output will be an XML based file. For an input sequence *Year_Purpose_Domain_Task_Sequence-ID*, the corresponding XML based output file should be named as *Site_System_P_Year_Purpose_Domain_Task_Sequence-ID_Run-ID.rdf*, correspondingly the contrastive submissions will be indicated as *Site_System_C_Year_Purpose_Domain_Task_Sequence-ID_Run-ID.rdf*

where,

- *Site* is a terse site ID
- *System* is a terse system name
- *Submission Type* can be (P, C), where P:Primary (by default) and C:Contrastive (must indicate which is contrastive when there are multiple submissions).
- *Year* specifies the year in which the evaluation would take place
- *Purpose* can be (Train, Test)
- *Domain* can be (Meetings, BNews, Surveillance, UAV)
- *Task* can be (HD, HT, PD, PT, VD, VT)
- *Sequence-ID* is the input sequence ID which can take values (1 ... N_{seq})
- *Run-ID* can take values (1 ... N_{run})

The description tags provided in this section are comprehensive to all tasks. However, only a subset of the tags relevant to each task are to be provided as specified in Sections 6-11. *Although the reference annotations and the system evaluations will be performed for only the I-frames, the systems will still be required to output the tags for all the frames in the sequence.*

The common and specific tags that should be provided by the systems are,

1. Filename of the video sequence.
2. Object ID.
3. Obox/Bbox specification (Obox if the box is oriented).
   (a) rotation in degrees (if Obox specified).
4. Frame Number/Framespan.

18 Required System Information

For each test run, a brief description of the system (algorithms, data, configuration) used to produce the system output must be provided along with your system output.). The system description information is to be provided in a file named: *Site_System_Year_Purpose_Domain_Task_Sequence-ID_Run-ID.sysinfo* and placed in the directory alongside the similarly-named directories containing your system output. This file is to be formatted as follows:

1. Site name
2. System Identifier/Name and version
3. Submitter (contact Name and email)

4. System Description:
   (a) Overview (high-level overview of system approach and configuration)
   (b) Features (description of pertinent system features)
   (c) Relationship to other runs (if this a comparative experiment, what other runs are related)
   (d) Configuration (particular configuration for this run)
   (e) Training (what training data was used and how was it employed)
   (f) Source Data Processing (how was the test data processed)
   (g) Equipment (what hardware was used, # of processors, type of processor, real and virtual memory, OS)
   (h) Processing Speed (what is the Speed Factor for this run as defined in Section 18.1)
   (i) Notes (any other notes regarding this system/run)
5. References: [list pertinent references]

18.1 Processing Speed Computation

The processing speed for each system run should be calculated as specified below and cited in the System Information file for the experiment. These are compulsory details that have to be reported in the system description for each submitted run.

18.1.1 Total Processing Time (TPT)

The time to be calculated is the Total Processing Time (TPT) that it takes to process all parallel streams of recorded video provided (including ALL I/O) on a single CPU. TPT represents the time a system would take to process the recorded video input and produce the specified meta-data output as measured by a stopwatch. So that research systems that aren't completely pipelined aren't penalized, the "stopwatch" may be stopped between (batch) processes.

Note that TPT may exclude time to "warm up" the system prior to loading the test recordings (e.g., loading models into memory.)

18.1.2 Source Signal Duration (SSD)

In order to calculate the real-time factor, the duration of the source signal recording must be determined. The source signal duration (SSD) is the actual recording time for the video audio used in the experiment. This time is stream-independent and should be calculated across all video streams for multi-view recordings. It is therefore the wall-clock duration of the period of recording (even if multiple simultaneous recordings were used).

18.1.3 Speed Factor (SF) Computation

The speed factor (SF) (also known as "X" and "times-real-time") is calculated as follows:

$$SF = \frac{TPT}{SSD}$$

For example, a 1-hour news broadcast processed in 10 hours would have a SF of 10. And 5 minutes of surveillance video collected on 2 cameras simultaneously each processed in 30 minutes would have an SF of 12.

18.2 Reporting Your Processing Speed Information

Although we encourage you to break out your processing time components into as much detail as you like, you should minimally report the above information in the system description for each of your submitted experiments in the form:

- TPT = <FLOAT>
- SSD = <FLOAT>
- SF = <FLOAT>

19 Submission Instructions

The system output XML files along with the corresponding System Information Files are to be *tar-ed* and then *gzipped*. For example, if the input sequences considered are 2005_Test_BNews_TDEng_1, 2005_Test_BNews_TDEng_2 and 2005_Test_BNews_TDEng_3, then

- The algorithm will use these sequences and output its results into a single XML file for each sequence in the same corresponding directory. Output file name should follow the file naming protocol presented in Section 17.2.
- Assume that the current working directory has all the sequences that the algorithm output is expected, all the XML files can then be compressed into a single file for submission by using the command,
  *tar -cvf Site_System_2005_Test_BNews_TDEng_Run-ID.tar *.rdf* then,
  *tar -rvf Site_System_2005_Test_BNews_TDEng_Run-ID.tar *.sysinfo* followed by
  gzip -9 Site_System_2005_Test_BNews_TDEng_Run-ID.tar which results in the file *Site_System_2005_Test_BNews_TDEng_Run-ID.tar.gz*.
- You can upload your file at this NIST site, details of which are listed below:

  1. ftp jaguar.ncsl.nist.gov
  2. log on with ”anonymous” passwd: ”your email address”
  3. cd /incoming/vace-eval

  Once you upload the file please do e-mail the person involved the specific details (filename, filesize)

  PS: As a security measure, you will not be able to see the file(s) after you upload it, so the only confirmation you will get is an ACK by e-mail.

20 Score Reporting

The primary measures that are reported for the detection is the SFDA with the appropriate thresholds (non-binary approach) or the SFDA-D depending on the specific task. For a ROC like analysis we will use the binary thresholded approach. In addition to these measures, we also report the missed detect and false alarms errors normalized with respect to the number of evaluation frames (I-frames).

The primary measures that are reported for the tracking is the ATA-T (non-binary threshold approach) or the ATA-D depending on the specific task. For a ROC like analysis again, we will use the binary thresholded approach (thresholds ranging from 0.1 to 0.9 with increments of 0.1). In addition to these measures, we will also report the missed detect and false alarm errors.

All these scores can be obtained by using the USF-DATE (USF-Detection And Tracking Evaluation) scoring package. The appropriate setting are described in the README file of the scoring package.

The exact flag settings used while scoring the SFDA-T and ATA-T measures on the system outputs are:

```
<prompt> usf_date GT_File SO_File Face -sfdat 0.2 -stdat 0.2
```

The exact flag settings used while scoring the outputs and primarily getting out the false alarms and missed detect errors are as given below:

```
<prompt> usf_date GT_File SO_File Face -sfdat 0.2 -stdat 0.2 -binthres
```

References

[1] D. Doermann and D. Mihalcik. Tools and techniques for video performance evaluation. In *ICPR*, volume 4, pages 167–170, 2000.

[2] M. L. Fredman and R. E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of ACM*, 34(3):596–615, Jul 1987.

[3] J. R. Munkres. Algorithms for the assignment and transportation problems. *J. SIAM*, 5:32–38, 1957.

[4] H. Raju and S. Prasad. Annotation Guidelines for Video Analysis and Content Extraction (VACE-II). Annotation Guidelines Document.

A APPENDIX: Matching Strategies

Assume that there are N ground truth objects and M detected objects. There needs to be a best possible match between these objects in a global sense. A brute force algorithm will have an exponential complexity, a result of having to try out all possible combination of matches (n!). However, this is a standard optimization problem and there are standard techniques to get the optimal match. The matching is generated with the constraint that the sum of the chosen function of the matched pairs is minimized or maximized as the case may be. In usual assignment problems, the number of objects in both cases are equal, i.e, when $N = M$. This is not a requirement and unequal number of objects can also be matched.

	DT_1	DT_2	$\ldots$	DT_M
GT_1	x			
GT_2				x
$\vdots$				
GT_N		x		

There are many variations of the basic Hungarian strategy [3] most of which exploit constraints from specific problem domains they deal with. The algorithm has a series of steps which is followed iteratively and has a polynomial time complexity, specifically some implementations have $O(N^3)$. Faster implementations have been known to exist and have the current best bound to be at $O(N^2 logN + NM)$ [2]. In our case, the matrix to be matched is most likely sparse and this fact could be taken advantage of by implementing a hash function for mapping sub-inputs from the whole set of inputs.